FINAL REPORT

# DEVELOPMENT OF NON-LINEAR FINITE ELEMENT COMPUTER CODE

To

California Institute of Technology

Jet Propulsion Laboratory

By

Eric B. Becker

Trent Miller

Contract No. 956442
Under NASA Contract NAS7-100
Task Order No. RE-182/213/210

September 1985

# ABSTRACT

Recent work by Peng ("Nonlinear Multiaxial Finite Deformation Investigation of Solid Propellants", S.T.T. Peng, AFRDL. TR-85-036) has shown that the use of separable symmetric functions of the principal stretches can adequately describe the response of certain propellant materials and, further, that a data reduction scheme given by him gives a convenient way of obtaining the values of the functions from experimental data. Based on Peng's representation of the energy, a computational scheme has been developed that allows finite element analysis of boundary value problems of arbitrary shape and loading.

The computational procedure has been implemental in a three-dimensional finite element code, TEXLESP-S, which is documented in this report.

## Table of Contents

## 1.0 INTRODUCTION

The finite element analysis of large elastic strain problems has been reported by many researchers, see for example, Oden [1], Cescotto and Fonder [2], Aly [3], Miller [4] Hubbitt et al. [5] and Becker et al. [6]. The constitutive relations employed in all of these works is a strain energy density which is taken as a function of the principal invariants of strain (technically, of the left Cauchy-Green deformation tensor ). No doubt the fondness exhibited toward this approach dates from the early work of Rivlin, in which it was shown that such forms of the energy function are acceptable for any isotropic hyperelastic material. In fact, most published reports of hyperelastic solutions employ a very simple form of the energy that was introduced by Mooney and has come to be known as the Mooney-Rivlin energy function. It is well known that this form of the energy describes the behavior of real rubber over only a very limitted range of deformations, but, due to its simple form (only two constants are required to specify the Mooney material) or, perhaps, due to the association with Rivlins fundamental work in rubber elasticity, the Mooney-Rivlin function continues to be used extensively. It is worth noting that the simplicity of form, while extremely valuable in the construction of analytical solutions to boundary value problems, is of but minimal significance in finite element work.

Viewed in the large, the stress analysis of any problem involves not only the finite element calculations used to satisfy equilibrium and boundary conditions but also the determination of a constitutive relation applicable to the material.

The characterization problem includes determining, from experimentally obtained data, both the mathematical form of the energy function and the value of the parameters ("material constants") embedded in these forms.

While theoretically convenient, the strain invariants are far from ideal choices of constitutive variables when viewed from the point of view of experimental determination of the constitutive form. Literally dozens of forms of the energy function, with strain invariants as arguments, have been proposed in the literature. The rationales for these are empirical rather than physical - material science considerations offer no guide to the form. The most common form is a polynomial in the strain invariants with the coefficients determined by least squares fitting to whatever data are available. Since any reasonable function can be approximated by a polynomial, so the argument goes, this simplistic approach should, in principle, be adequate. Unfortunately, polynomial interpolation invariably produces oscillatory behavior in the function and, especially, in its derivatives. Since it is the first and second derivatives of the energy that are required in any computational procedure, the high order polynomials required to give adequate fits to real material behavior over large ranges of deformation do not work well in practice.

The use of principal stretches as arguments of the energy function has been proposed by, for example Valanis and Landell [7], Ogden [8] and Peng [9]. When the stretches are chosen as constitutive variables, an assumption can be made that reduces the difficulty of the characterization problem. Using the assumption of separability, Ogden, for example, has been able to fit a wide range of rubber deformations using fairly simple mathematical forms of the energy function. Ogden's characterization, however, requires the choice of some parameters on an intuitive basis.

The approach to material characterization taken here differs from previously reported approaches in that no mathematical form of the energy function is postulated. Since only the values of the derivatives of the energy and not form of the function, are needed for computation, we avoid the restrictions inherent in the choice of a particular mathematical form. Working directly with interpolated experimental data we calculate points on a curve that defines the material response. These points are supplied as input data to the finite element analysis code. Our procedure, thereby, provides the most direct and straight forward use of experimental data for the solution of boundary value problems. The details of this procedure are given in Section 2.

The finite element code, TEXLESP-S developed in the present effort utilizes components of the code TEXGAP3D, which is described in reference [10]. The modelling capability is identical to that of TEXGAP3D and almost all of the data are identical. The hyperelastic calculations are based on proceedures developed by Aly in reference [3], but modified to allow the use of the separable energy function of principal stretches. The use of TEXLESP-S is described in Section 3.

## 2.0 THEORETICAL DEVELOPMENT

The theory of large elastic deformations was essentially set forth by Rivlin in the late 1940's. Notations have changed and the use of finite element techniques has shifted the emphasis but the theoretical foundations are unchanged. The major change in emphasis is the use of variational principles (principle of virtual work, for example) to satisfy equilibrium requirements and the accompanying use of Lagrange multiplier methods to accommodate the incompressibility condition.

In the following sections we review the notation and description of deformation, Section 2.1; the constitutive relations on which this work is based, Section 2.2; the variational formulation of the boundary value problems, Section 2.3 and the finite element implementation of the theory, Section 2.4.

## 2.1 Kinematics of Deformation

Let $\underset{\sim}{X}$ denote the position of a material particle in the undeformed (and unstressed) reference configuration of a body whose deformation is to be studied. At some later time, $t$ , as the body is deformed by applied loads, the particle originally at $\underset{\sim}{X}$ moves to a new position $\underset{\sim}{x}$ . The problem to be solved is the determination for a fixed time, the value of $\underset{\sim}{x}$ for every particle, $\underset{\sim}{X}$ , in the body consistent with the requirement that each part of the body be in equilibrium with applied loads. We consider only slow variation in the loads, so that inertia is not important, and elastic behavior, so that the history of deformation is not important. Thus, time appears in all equations only as a parameter identifying which set of loads and deformations are being studied. It will be convenient to replace time, then, with another parameter, $\rho$ , which we shall call the load factor. The load factor will vary, under control of the analyst, from a value of zero in the reference configuration through whatever positive values are of interest. When the load factor has a value of unity, the loads acting on the body will have what we shall call their nominal values. We note that since all loads are scaled by the same parameter we are restricting attention to proportional loading.

The motion of the body is described mathematically by the function (unknown, a priori)

$$\underset{\sim}{x} = \underset{\sim}{x}(\underset{\sim}{X}, \rho) \tag{1}$$

from which we shall generally omit the dependence on load factor $\rho$ .

In an elastic material, the stress at a particle $\underset{\sim}{X}$ depends only on the difference between the shape of an infinitesimally small portion of the body around $\underset{\sim}{X}$ and the shape of this same portion in the reference configuration. If $d\underset{\sim}{S}$ denotes a line element in the reference configuration (at particle $\underset{\sim}{X}$) and $d\underset{\sim}{s}$ denotes the deformed configuration of this element, then the relation between these is given by

$$d\underset{\sim}{s} = \underset{\sim}{F}\, d\underset{\sim}{S} \tag{2}$$

where the deformation gradient $\underset{\sim}{F}$ is calculated as

$$\underset{\sim}{F} = \frac{\partial \underset{\sim}{x}}{\partial \underset{\sim}{X}} \tag{3}$$

The determinant of the deformation gradient, $\underset{\sim}{F}$ , plays an important role in the kinematics of deformation. Direct calculation shows that the ratio of the volume contained in a deformed material region $dv$ to the volume of the same material in the reference configuration dV is given by

$$\frac{dv}{dV} = \det|\underset{\sim}{F}| = J \qquad (4)$$

This determinant, denoted by J , is often callled the Jacobian determinant of the deformation or, simply, the Jacobian. Clearly, for all physically acceptable deformations $J > 0$ and for isochoric (i.e., volume preserving) deformations as must occur in incompressible materials, $J = 1$ .

Since the change in shape of the infinitesimal part of the body surrounding the particle $\underset{\sim}{X}$ is completely determined by the knowledge of the changes of all the line segments emanating from the particle, it is clear that $\underset{\sim}{F}$ contains all of the information required. In fact, $\underset{\sim}{F}$ contains not only the description of changes in shape of the neighborhood of $\underset{\sim}{X}$ , but also changes in its orientation, since a rigid body rotation of the neighborhood would change each $d\underset{\sim}{S}$ into a $d\underset{\sim}{s}$ with the same length but a different direction. A useful theorem of linear algebra (the polar decomposition theorem) assures us that $\underset{\sim}{F}$ can be factored in such a way that the rigid body rotation and stretching parts of the deformation are separate, Thus,

$$\underset{\sim}{F} = \underset{\sim}{R}\underset{\sim}{U} \qquad (5)$$

Since rigid body rotation does not affect the stress, we are interested in the stretch tensor $\underset{\sim}{U}$ . Although the polar decomposition theorem assures the existence of $\underset{\sim}{U}$ , it does not offer a convenient way to determine it. But since $\underset{\sim}{R}$ represents a rigid rotation, $\underset{\sim}{R}^{-1} = \underset{\sim}{R}^T$ , the tensor $\underset{\sim}{C}$ , defined by (6), depends only on the stretches.

$$\underset{\sim}{C} = \underset{\sim}{F}^T\underset{\sim}{F} = \underset{\sim}{U}^T\underset{\sim}{R}^T\underset{\sim}{R}U = \underset{\sim}{U}^T\underset{\sim}{U} \equiv \underset{\sim}{U}^2 \qquad (6)$$

Clearly, $\underset{\sim}{C}$ is a useful measure of the stress producing deformation of the neighborhood of the material surrounding the particle $\underset{\sim}{X}$ . Another measure of this deformation (not used here) is the Green strain tensor, defined as

$$\underset{\sim}{E} = \frac{1}{2}(\underset{\sim}{C} - \underset{\sim}{I})$$

The tensor C is symmetric, and, therefore, has real principal values. If these values are denoted, say, $\mu_i$ , $i = 1, 2, 3$, then the characteristic equation of $\underset{\sim}{C}$ is

$$\mu^3 - I_1\mu^2 + I_2\mu^2 - J^2 = 0 \qquad (7)$$

In (7), $I_1$, $I_2$ and $J^2$ are the three principal invariants of $\underset{\sim}{C}$ . The third of these is written here as $I_3 = J^2$ emphasizing the fact that it is equal to the square of the Jacobian.

Since $J > 0$ , the tensor $\underset{\sim}{C}$ is positive definite, i.e., $\mu_i > 0 \quad i = 1, 2, 3$ . Recalling the last definition in (6), we denote the eigenvalues of $\underset{\sim}{C}$ by $\lambda^2$ rather than $\mu$ . The interpretation of the $\lambda_i$ is that they are the principal values of the stretch tensor $\underset{\sim}{U}$ or, simply, the principal stretches. Rewriting (7) as

$$(\lambda^2)^3 - I_1(\lambda^2)^2 + I_2(\lambda^2) - I_3 = 0 \qquad (8)$$

clearly shows that a knowledge of the three principal invariants $I_1$, $I_2$, and $I_3$ implies a knowledge of the three principal stretches $\lambda_1$, $\lambda_2$, $\lambda_3$ and vice versa. Thus, either set of quantities can, in principle, be used to describe the stress-producing aspect of a deformation.

The geometric interpretation of the tensors $\underset{\sim}{F}$ , $\underset{\sim}{R}$ and $\underset{\sim}{U}$ and of the principal stretches are shown, for a two-dimensional case, in Figure 1. The circular neighborhood, N , surrounding $\underset{\sim}{X}$ is shown in Figure 1a. Three line elements through $\underset{\sim}{X}$ are shown. These are labeled 1, 2 and 3. Lines 1 and 2 are in the principal directions of the stretch tensor $\underset{\sim}{U}$ , while 3 is simply another, arbitrarily chosen, line segment. The deformed configuration of the neighborhood, n , is depicted in Figure 1b. We note that, in general, all of the line elements in N have been rotated and stretched (or compressed). This is the result of the deformation described by (2), and we say that $\underset{\sim}{F}$ carries N into n . If the diameter of N is taken as unity, then the lengths of linear segments 1 and 2 in configuration n (Fig. 1b) are $\lambda_1$ and $\lambda_2$ . These are the values of stretch (ratio of deformed to undeformed length) for those line elements that get stretched the most and least of all line segments through the point $\underset{\sim}{X}$ .

Figure 1c shows the configuration of neighborhood $N$ that would be produced purely by the stretching part of $\underset{\sim}{F}$ , had no rigid body rotation occurred. If we denote this $\bar{n}$ , we say that $\underset{\sim}{U}$ carries $N$ into $\bar{n}$ while $\underset{\sim}{R}$ carries $\bar{n}$ into $n$ . We note that there is no rotation of the principal line elements, 1 and 2, produced by $\underset{\sim}{U}$ and that there are no length changes of any line elements produced by $\underset{\sim}{R}$ .

## 2.2 Constitutive Equations

We consider only isotropic, incompressible hyperelastic materials. For this class of materials, the stresses are determined, to within a hydrostatic pressure, from a scalar function called the strain energy density function or, simply, the energy function. In most of the published work on rubber elasticity, the energy is written as a function of the principal invariants as

$$U = U(I_1, I_2, I_3) \qquad (9)$$

The Cauchy stress (traction per unit of area in the deformed configuration) is, in terms of the function $U(I_1, I_2, I_3)$

$$\underset{\sim}{\sigma} = 2[(U_{,1} + I_1 U_{,2})\underset{\sim}{B} - U_{,2}\underset{\sim}{B}^2] + p\underset{\sim}{I} \qquad (10)$$

where $U_{,i} \equiv \dfrac{\partial U}{\partial I_i}$

$\underset{\sim}{B} \equiv \underset{\sim}{F}\underset{\sim}{F}^T$

$\underset{\sim}{I}$ is the identity

$p$ is the hydrostatic pressure

In this work, we choose to express the energy function, not explicitly as a function of the invariants, but, rather, following Valanis and Landel, as a separable function of the principal stretches. Thus, we write

$$U = w(\lambda_1) + w(\lambda_2) + w(\lambda_3) \qquad (11)$$

The Cauchy stress for an incompressible material for which (11) holds is given by

$$\underset{\sim}{\sigma} = \sum_{i=1}^{3} \lambda_i \, w'(\lambda_i) \, \underset{\sim}{n}^i \times \underset{\sim}{n}^i + p\underset{\sim}{I} \qquad (12)$$

In (12), $\underset{\sim}{n}^i$ is the unit vector in the direction of the principal stretch $\lambda_i$ .

For the solution of the equilibrium equations (as will be seen in section 2.4), we shall need the first and second derivatives of the energy function with respect to the invariants. If we were using the form (9), this calculation could be made easily and explicitly, but when using (11) as our constitutive assumption, we proceed as follows.

Let

$$U,_k \equiv \frac{\partial U}{\partial I_k} \quad \text{and} \quad U,_{k\ell} \equiv \frac{\partial^2 U}{\partial I_k \partial I_\ell} \qquad (13)$$

denote the derivatives of the energy with respect to the three invariants $I_1$, $I_2$ and $I_3 \equiv J^2$ . Also, let

$$I_{k,i} \equiv \frac{\partial I_k}{\partial \lambda_i} \quad \text{and} \quad I_{k,ij} \equiv \frac{\partial^2 I_k}{\partial \lambda_i \partial \lambda_j} \qquad (14)$$

denote the derivatives of the invariants with respect to the principal stretches. The derivatives in (14) are readily calculated from

$$I_1 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$$
$$I_2 = \lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_1^2\lambda_3^2 \qquad (15)$$
$$I_3 \equiv J^2 = \lambda_1\lambda_2\lambda_3$$

and the results can be written concisely as

$$I_{1,i} = 2\lambda_i \qquad I_{1,ij} = 2\delta_{ij}$$
$$I_{2,i} = 2\lambda_i(I_1 - \lambda_i^2) \qquad I_{2,ij} = \begin{cases} 2(I_1 - \lambda_i^2) & i \neq j \\ 4\lambda_i\lambda_j & i = j \end{cases}$$
$$(16)$$
$$J_{,i} = J/\lambda_i \qquad J_{,ij} = \begin{cases} 0 & i = j \\ J/\lambda_i\lambda_j & i \neq j \end{cases}$$

Equating (9) and (11) and differentiating the result with respect to $\lambda_i$ gives three equations for the determination of $U_{,k}$

$$U_{,k}\, I_{k,i} = w'(\lambda_i) \quad i = 1, 2, 3 \qquad (17)$$

We note that for known values of the $\lambda_i$ the coefficients $I_{k,i}$ in (17) are evaluated using (16). For a given function $w(\lambda)$ we obtain the derivative $w'(\lambda_i)$ by the process described below. Thus, equations (17) are three linear algebraic equations that are solved for the required values of $U_{,k}$ .

The second derivatives $U_{,k\ell}$ are obtained in a way similar to that just described. Differentiating (17) this time with respect to $\lambda_j$ we obtain

$$U_{,k\ell}\, I_{k,i}\, I_{\ell,j} + U_{,k}\, I_{k,ij} = \left(w'(\lambda_i)\right)_{,j} \qquad i,j = 1,\ 2,\ 3 \tag{18}$$

Equations (18) are six linear algebraic equations for the determination of the values of $U_{,k\ell}$ . We note that the values of the first derivatives $U_{,k}$ must be found by solving (17) before equations (18) can be formed.

When two (or three) of the principal stretches are identical, then (17) and (18) are singular, and special care must be taken with the degenerate systems. This is an exceptional situation that almost never occurs in the solution of boundary value problems. TEXLESP-S contains adequate provisions to detect and to deal with the singular cases.

In our implementation, values of the function $w'(\lambda)$ are input at several values of $\lambda$ . A typical curve defined by these data is in Figure 57 of Peng's report [9]. Values of $w'$ must be calculated by interpolation for use in (17) and values of $w''$ must be calculated for use in (18). The functional values of $w'$ must be smooth so that $w''$ will be continuous for all values of $\lambda$ or else the tangent stiffness matrix (see eqs. (7) and (10) in Section 2.4) cannot be calculated. To assure smooth variation of $w'$ with respect to $\lambda$ , we fit the data with cubic splines (using Hermite polynomial interpolating functions between data points).

## 2.3 Finite Element Formulation

The finite element formulation is based on the principle of virtual work, which is written as

$$\int_{\Omega} \delta U \, dV = \int_{\Omega} f \cdot \delta x \, dV + \int_{\partial\Omega} t \cdot \delta x \, dS \qquad (19)$$

In (19), the notation is defined as follows:

- $\delta U$ is the variation of the energy function with respect to the current position $x$
- $f$ is the vector of body force densities
- $t$ is the vector of surface tractions
- $\delta x$ is the variation of current position
- $\Omega$ is the interior of the body in the reference configuration
- $\partial\Omega$ is the boundary of the body in the reference configuration.

We note that the boundary of the body is, in general composed of a portion, say $\partial\Omega_t$ , on which the tractions, $t$ , are given and a portion, $\partial\Omega_x$ , on which the deformed position fo the material is given (and on which the traction is unknown but on which the variation of position, $\delta x$ , vanishes). Thus, the region of integration of the surface integral can be taken simply as $\partial\Omega_t$ .

Satisfaction of (19) for all suitably smooth $\delta x$ guarantees equilibrium (in a variational or weak sense). In order that the deformation be isochoric (i.e., that the incompressibility of the material be satisfied), we must add to (19) the condition that $J - 1 = 0$. This is accomplished using Lagrange's method of multipliers. The result is the modified principle of virtual work

$$\int_{\Omega} \delta\left(U + p(J-1)\right)dV = \int_{\Omega} f\cdot\delta x dv + \int_{\partial\Omega_t} t\cdot\delta x dS \qquad (20)$$

Here, p, the Lagrange multiplier, is the hydrostatic pressure occurring in the constitutive equation (10). We rewrite the left-hand member (20) as

$$\int_{\Omega} \delta\bar{U}\ dV \qquad (21)$$

and note that the variation is with respect to $p$ as well as with respect to $x$ and that the function now depends on $J$.

The body is represented by an assemblage of finite elements. The geometry of each element is defined by the coordinates of the nodal points connected to it and by a set of functions called shape functions, denoted by $N^{\alpha}(s_i)$ $i = 1, 2, 3$. The parameters $s_i$ are called the parametric coordinates of the material particles contained in an element and can be thought of as describing the location of a point in a master element. Typically, the $s_i$ range over a simple interval such as $-1 \leq s_i \leq 1$.

In TEXLESP-S, the element library consists of 20 node bricks, 15 node triangular prisms and 10 node tetrahedra. The details of these elements can be found in, for example, the book by Zienciewicz [11] or the TEXGAP3D reference manual [10]. The shape functions in each type of element are quadratic functions of the $s_i$.

All integrals in the virtual work statement and its consequences are evaluated by numerical quadrature element-by-element. Thus, all calculations required in the following development are performed at certain fixed points within an element called integration points. Once the finite element discretization is made, there are no longer any functions of position (either known or unknown) and the only undetermined parameters are the nodal point values of position, $x$ , and pressure, $p$ .

Explicitly, the reference position of a point, the deformed position and the pressure at a point are given by the following

$$\begin{array}{lll} X_i = N^\alpha X_i^\alpha & i = 1, 2, 3 & \\ x_i = N^\alpha x_i^\alpha & \alpha = 1, 2, \ldots \text{ Node} & (22) \\ p = M^\beta p^\beta & \beta = 1, 2, 3, 4 & \end{array}$$

Similarly, the variation of the components of $\underset{\sim}{x}$ and of $p$ are given by

$$\delta x_i = N^\alpha \; \delta x_i^\alpha$$

$$\delta p = M^\beta \; \delta p^\beta \qquad (23)$$

For convenience in the following, we will let $\underset{\sim}{u}$ stand for the entire set of nodal quantities, including both $x_i^\alpha$ and $p^\beta$ where $\alpha$ ranges over all nodes in the model, and $\beta$ ranges from 1 to 4 for each element of the model. Similarly, other quantities with a sub-tilde, e.g., $\underset{\sim}{I}$ , will denote vectors with a corresponding number of components. Double sub-tildes will denote square matrices of compatible size.

When the finite element discretizations (22) and (23) are substituted into the modified principle of virtual work, the results can be written as

$$\delta \underset{\sim}{u}^T \underset{\sim}{I} = \delta \underset{\sim}{u}^T \underset{\sim}{F} \qquad (24)$$

in which

$$\underset{\sim}{I} = \frac{\partial}{\partial \underset{\sim}{u}} \int_\Omega \bar{U} dV$$

$$\underset{\sim}{F} = \int_\Omega f \cdot \underset{\sim}{N} dV + \int_{\partial\Omega} t \cdot \underset{\sim}{N} dS$$

The equilibrium equation (24), which is really a vector equation with one component for each degree of freedom in the problem, is nonlinear since $\underset{\sim}{I} = \underset{\sim}{I}(\underset{\sim}{u}, \underset{\sim}{p})$ is highly nonlinear. The solution of this system of equations is accomplished by means of incremental loading combined with

Newton iteration. As noted in Section 2.1, we consider proportional loading so that we can write $f = \rho\hat{f}$ and $t = \rho\hat{t}$ for the applied loads. Using these conventions in (25), we obtain the obvious modification of (24)

$$\underset{\sim}{I} = \rho\hat{\underset{\sim}{F}} \qquad (26)$$

for the equilibrium equations. The incremental loading procedure consists of proceeding from a value of $\rho$ at which (26) is satisfied to an incremented value, say $\rho + \Delta\rho$ , and attempting to satisfy (26) again. The first step in this consists of a linearization of (26) about the current value, i.e.,

$$\underset{\sim}{I}(\underset{\sim}{u} + \Delta\underset{\sim}{u}) = \underset{\sim}{I}(\underset{\sim}{u}) + \frac{\partial\underset{\sim}{I}(\underset{\sim}{u})}{\partial\underset{\sim}{u}}\Delta\underset{\sim}{u}$$

from which we obtain as the incremental equation

$$\frac{\partial\underset{\sim}{I}(\underset{\sim}{u})}{\partial\underset{\sim}{u}}\Delta\underset{\sim}{u} = (\rho + \Delta\rho)\hat{\underset{\sim}{F}} - \underset{\sim}{I}(\underset{\sim}{u}) \qquad (27)$$

This set of linear equations has as its coefficient matrix the tangent stiffness $\underset{\approx}{K}$ defined as

$$\underset{\approx}{K} = \frac{\partial\underset{\sim}{I}}{\partial\underset{\sim}{u}} \qquad (27)$$

We note that it is the evaluation of this matrix and of the vector $\underset{\sim}{I}(\underset{\sim}{u})$ on the right-hand side of (26) that requires the bulk of the calculations in TEXLESP-S.

In the first step of solving for a new load increment $\underset{\approx}{K}$ and $\underset{\sim}{I}$ are evaluated at the previous configuration and the change in position $\Delta\underset{\sim}{u}$ is found by solving (26). This solution is accomplished in TEXLESP-S using a frontal elimination routine. In general, the incremented state $\underset{\sim}{u}^1 = \underset{\sim}{u} + \Delta\underset{\sim}{u}$ will not satisfy equilibrium, that is to say, the residual

$$\underset{\sim}{R}^1 = (\rho + \Delta\rho)\hat{\underset{\sim}{F}} - \underset{\sim}{I}(\underset{\sim}{u}^1)$$

will not be zero. To improve on the solution, we linearize again -- this time about the configuration $\underset{\sim}{u}^1$ . This process is

$$\underset{\approx}{K}^i d\underset{\sim}{u}^{i+1} = \underset{\sim}{R}^i$$

$$\underset{\sim}{u}^{i+1} = \underset{\sim}{u}^i + d\underset{\sim}{u}^{i+1}$$

and is repeated until the correction $d\underset{\sim}{u}$ becomes less than some acceptable tolerance.

In practice, the user of TEXLESP-S specifies a sequence of load factors, say $\rho_1$ , $\rho_2$ , $\rho_3$ ... $\rho_n$ , and a tolerance for the convergence of the $d\underset{\sim}{u}$ . For each load step, the solution, $\underset{\sim}{u}$ , is saved on file. Post-processing, i.e., stress calculation, printing and plotting, can be done for any converged load step. The solution procedure can be continued by restarting the code and specifying additional load factors.

The details of the calculation of $\underset{\sim}{I}$ and $\underset{\approx}{K}$ are contained in the Appendix.

## 3.0 TEXLESP-S USERS GUIDE

The finite element code TEXLESP-S solves equilibrium problems for three-dimensional incompressible hyperelastic bodies. The constitutive equations are given by specifying the energy function as either a polynomial in the strain invariants or a separable function of the principal stretches. In the latter case, the material data are input as points on the $w'$ versus $\lambda$ curve.

The preprocessing and post-processing functions of TEXLESP-S have been adapted from the code TEXGAP3D for the solution of three-dimensional linear elastic problems. Insofar as possible, the differences between these codes have been made transparent to the user. That is to say, all modeling and post-processing data are identical in the two codes. Only the material property descriptions and commands directing the solution procedures are different. Consequently, the TEXGAP3D users manual can be used, verbatim, for all descriptions of input formats, mesh generation commands, element definitions, boundary condition specifications and post-processing commands.

This users' guide contains an overview of the structure of the data deck (Section 3.1); a description of those data commands that are different from TEXGAP3D data (section 3.2); and a set of example problems which illustrate the use of the code (Section 3.3).

## 3.1 Structure of Code and Data Deck

Data deck structures for TEXLESP-S and TEXGAP3D are essentially the same. In each code, there are data required to

a) set up the model

b) solve the problem

c) post-process the solution.

While these are typically all contained in a TEXGAP3D run, they may occur in separate runs for TEXLESP-S. Using the restart capability in TEXLESP-S is often desirable and requires some understanding of the code structure.

Figure 2 shows a large-scale flow diagram of TEXLESP-S. It can be noted that any run begins with the main routine. From the main routine, control can be directed to

a) SETUP - to define a model

b) SOLVE - to calculate solutions for various load increments

c) POST - to calculate print and plot stresses, strains and/or displacements

d) RESTART - to resume execution of problems begun on a previous run.

It will be noted that the file TAPE12 is used by all modules. This file contains all data that describe the model, i.e., nodal point coordinates, element definitions and boundary condition specifications, as well as the solutions calculated at each load step. Data are written to TAPE12 by SETUP and by SOLVE. Any run which calls these modules modifies TAPE12 and, if subsequent processing of the job is to be performed, the updated version of this file must be saved.

TAPE18 is a file that contains the material data for hyperelastic materials whose energy function is written as a separable function of the principal stretches. These data are read by TEXLESP-S in the SOLVE and POST modules but are not modified. This file must be attached to the job when SOLVE or POST are to be used. The format of TAPE18 is such that the following statements can be used to read it.

```
      READ (18,2010)N
      READ (18,2020)(ALAM(I), I = 1,N)
      READ (18,2020)(WP(I), I = 1,N)
 2010 FORMAT(I5)
 2020 FORMAT(6F11.3)
```

The data read by these statements are:

N the number of points on the $w'$ vs $\lambda$ curve

ALAM(I) values of $\lambda$ , in ascending order

WP(I) values of $w'$ .

Data Deck Structure. The structure of data decks for TEXLESP-S will vary according to the nature of the job being processed. In Figure 3, we show the most general structure, i.e., that which is used when a problem is to be solved in a single run. The data cards that are given explicitly are similar for all such jobs. Data that are problem-dependent are not given explicitly in Figure 3.

Figure 4 shows the structure of a data deck that would be used to continue a previously started solution and to perform post-processing on it.

## 3.2 Description of TEXLESP-S Commands

Only those data cards that differ from the corresponding TEXGAP3D data are described here. For complete descriptions of the other data, see the TEXGAP3D Users' Manual [10]. The following descriptions explain all data cards that differ from those of TEXLESP3D.

### 3.2.1 Integration control card.

SETINT

This card, which should occur before the SETUP card changes the default number of integration points from 14 to 8. A significant saving in computer time can be realized by the use of the SETINT card.

### 3.2.2 Material property definition cards

RUBBERS, name, no.

This card defines the material, "name," as material number, "no" and specifies that it is a rubber elastic material whose energy function is defined by the data on TAPE18.

Note that only one such material can be used in a problem.

RUBBER I, name, no, n, C10, C01, C20, C11, C02
C30, C21, C12, C03

This card defines material, name, as material number "no." The energy function for the material is defined in terms of the strain invariants in the following way, depending on the value of n .

$\underline{n = 1}$ $U_1 = C10(I_1-3) + C01(I_2-3)$

$\underline{n = 2}$ $U_2 = U_1 + C20(I_1-3)^2 + C11(I_1-3)(I_2-3) + C20(I_2-3)^2$

$\underline{n = 3}$ $U_3 = U_2 + C30(I_1-3)^3 + C21(I_1-3)^2(I_2-3)$
$+ C21(I_1-3)(I_2-3)^2 + (C03(I_2-3)^3$

These are the only two material types available in TEXLESP-S.

### 3.2.3 Grid Generation

The grid generation commands in TEXLESP-S are identical to those in TEXGAP3D. See Section 2.4 of the TEXGAP3D Users' Manual for descriptions of these commands.

### 3.2.4 Element Definition and Boundary Condition Specification

The element library and element generation commands are identical in TEXLESP-S and TEXGAP3D except that the singular WEDGE elements are not used in TEXLESP-S.

All of the boundary condition commands in TEXGAP3D are available in TEXLESP-S. In addition, a CLAMP command, described below, has been added to TEXLESP-S.

The element and boundary condition specification (with the exceptions noted above) are described in Section 2.5 of the TEXGAP3D users' manual.

```
BC, CLAMP, i, j, k, nside, nset, vx, vy, vz
```

This boundary condition specifies that all of the nodes on side, "nside" of element "i, j, k" have specified displacements. The values of these displacements are vx, vy and vz in the x, y and z directions respectively. Each different set of vx, vy, vz is given a number, "nset," the first time it is used. When a given set of applied displacements is used subsequently only the set number (n set) need be repeated. See example data set 1 for use of the CLAMP command.

### 3.2.5 Solution commands

In TEXGAP3D, the command, SOLVE, has no parameters, since only a linear solution is produced. In TEXLESP-S, there are parameters on the SOLVE card that control the incremental loading and Newton iteration processes.

SOLVE, jprint, iter, tol, $\rho_1$, $\rho_2$, $\rho_3$, ..., $\rho_m$

The SOLVE command causes the solution of the equilibrium equations to be performed using the sequence of load factors $\rho_1, \rho_2, \rho_3, \ldots, \rho_m$. Each solution is recorded on TAPE12. If no previous solutions have been stored on TAPE12, then these are the first n solutions. On the other hand, if a previous run of the problem has produced solutions and the current run is a RESTART run, then the sequence of solutions produced is recorded following the solution at which the restart was made. See example problem 2 for an illustration of this.

The parameter, jprint, controls the amount of print produced during the solution. Increasing values of jprint produce increasing amounts of output. A value of 1 produces the convergence summary table and is recommended.

The maximum number of iterations allowed in a load step is equal to "iter." The default value, which is 10, is recommended.

The parameter, "tol," defines the tolerance to which the changes in displacement are compared when determining whether the Newton iterations have converged. The default value is $10^{-4}$ which is small enough for nearly all purposes. The convergence check determines that

$$du_i < tol * |u_i|$$

for each component of displacement in the model. Here, $u_i$ is the value of the displacement component and $du_i$ is the correction to that displacement produced by the Newton iteration.

A summary of the convergence tests is printed for each load step. If the iterations should fail to converge at any load step, all of the solutions up through the previous load step are stored on TAPE12 so that restarting can be done from that point.

### 3.2.6 Post Processing

The post processing in TEXLESP-S uses commands that, with two exceptions, are the same as those used in TEXGAP3D. All plotting and stress calculation commands are as described in Section 2.7 of the TEXGAP3D users' manual

```
POST, nstep
```

The POST command in TEXLESP-S contains the parameter "nstep" which specifies the load step number of the solution to be processed.

```
REACTION, xmin, ymin, zmin, xmax, ymax, zmax,
          imin, jmin, kmin, imax, jmax, kmax
```

The REACTION command causes the reactions at all nodes within the specified (x,y,z) bounds and the specified (i,j,k) bounds to be printed. Either or both sets of bounds can be omitted. If no bounds are specified, then all nodes in the model are processed.

The reactions that are printed are actually the unbalanced nodal point forces (i.e., the difference between specified applied loads and the calculated internal forces). When equilibrium is exactly satisfied at a node, the reactions will be equal to zero, unless the node has had displacements specified. When displacements are specified, then the actual reaction furnished by the support on the model is the calculated value.

The vector sum of all of the reactions in the specified part of the model is also printed. The REACTION command is, thus, very useful in determining load-deflection data.

## 3.3 Example Problems

The example problems presented in this section are intended to illustrate the application of TEXLESP-S in a typical application. Descriptions of the input data for several runs associated with the problem are described below.

The problem to be analyzed is the stretching of a thin biaxial test specimen with a rail cross section. The specimen is shown in Figure 5. This is the same specimen for which results are given in Peng's report [9] (see Fig. A-2 in [9]).

The separate runs that are described below are:

Run 1 - Generation and plotting of grid

Run 2 - Analysis of stretching up to 20%

Run 3 - Analysis of stretching from 20% to 90%

Run 4 - Calculation of reactions

Run 5 - Stress calculation and deformed shape plotting.

It is desirable, although not necessary, to perform nonlinear analysis in several stages using the restart capability of TEXLESP-S. The use of several separate runs allows the user to check intermediate results to make sure, for example, that the grid is error free and that convergence of the nonlinear solution is taking place.

For runs 2 through 6, TAPE12 from the prior runs and TAPE18 containing material properties are required.

Run 1: Generation and Plotting of a Grid

The first step in generating a grid for TEXLESP-S is to sketch the grid and to assign (I, J, K) node numbers. Figure 6 shows such a sketch of one-eighth of the specimen. In this grid, the I, J, K directions are in the x, y, z directions respectively. Since the elements are ordered, for solution purposes, by varying I then J then K it is especially important to let I vary in the direction of fewest elements (one or two in this case) and K in the "long" direction (5 elements in this case). It will be noted that several nodes have been identified on the sketch.

The data deck for this run is given in Figure 7. Following the title card, there is a SETINT command (Section 3.2.1). Then the SETUP command calls the pre-processing overlay.

The material model in this problem is given by the $w'$ versus $\dot{\lambda}$ data which are contained on TAPE18. The RUBBERS card specifies this and identifies the material as material number 1.

The grid generation proceeds by defining all of the nodes on the plane $z = 0$ which is also $K = 1$ . FACE, ARC, POINT and CONNECT commands are used to define these points. After all of the $K = 1$ points have been defined, the NORMAL command is used to define other K planes. After all nodes have been defined, the element definition begins. The elements in this problem are BRICKH and PRISMH elements. The looping features of TEXLESP-S are used in defining the elements.

The boundary conditions on the one-eighth model are

(a) Symmetry (SLOPE) on the plane $y = 0$ which is face 5 of elements (1, 1, K).

(b) Symmetry (SLOPE) on the plane $x = 0$ which is face 4 of elements (1, 1, K) and (1, 3, K).

(c) Symmetry (SLOPE) on the plane $z = 3.75$ which is face 3 of elements (1, 1, 9) and (1, 3, 9) and face 4 of the PRISMH, (3, 3, 9).

(d) All displacements specified (CLAMP) on the plane $y = .75$ . This is face 2 of elements (1, 3, K) and face 1 of the PRISMH elements (3, 3, K). The values of displacements specified in the CLAMP command (Section 3.2.4) are $u = 0$, $v = .75$, $w = 0$. Note that this corresponds to a stretch of 100%. This value would be achieved by a load factor of 1.0.

After the elements and boundary conditions have been specified, the preprocessing phase is ended.

In order to examine the grid generation results before attempting an analysis, this run ends with a call to the post processor to plot the grid. The plot produced by this call is shown in Figure 8. In addition to the plot, a large amount of printed output is generated, from which detailed checking of nodal coordinates, element connectivities and boundary condition specification can be done.

The data generated by this run are written on TAPE12 so that subsequent runs need not repeat the grid generation process.

### Run 2: Analysis of Stretching Up to 20%

Although final results may be desired only for a couple of values of stretch, we choose to increment the load rather slowly at first. This strategy has the benefit of producing more complete load-deflection data. More significantly, it is often the case that smaller load steps are necessary in the early stages of loading in order to achieve convergence. In the present example, for instance, a load increment from 0 to 20% will not converge while steps from 15% to 20% as well as from 20% to 40% do converge.

Examination of the load-deflection curve for this example, Figure 9, indicates a rapid change in behavior around a stretch of 10%. It is this behavior that is reflected in the need for initially small load steps.

The data set for this run is shown in Figure 10. This run requires attachment of TAPE12 from Run 1 and TAPE18 containing the material data. The only output from this run describes the convergence of each load step. The resulting solutions are accumulated on TAPE12.

Run 3: Analysis of Stretching from 20% to 80%

After having determined, from the results of Run 2, that the solution is converging satisfactorily, this run is used to continue the solution for subsequent load steps. Note that the restart card contains the specification of the load step beyond which the next SOLVE command will calculate solutions. TAPE12 and TAPE18 are required.

Run 4: Calculation of Reactions

In this run, the code is restarted and the solutions at load steps 1, 2 and 3 are post processed. The REACTION command produces node-by-node reactions for all nodes within the xyz bounds specified. These are not as useful as the total resultants that are the sum of these nodal reactions. The region specified in this run includes the plane $y = .75$ but not the plane $y = 0$. Thus, the $y$ reaction printed is the total applied load on one fourth of the top grip. The total applied load is, of course, four times this value.

Figure 9 shows the load deflection data plotted by a separate plotting program.

Run 5: Stress Calculation and Deformed Shape Plotting

To obtain values of displacement, stress and/or strain at various locations in the model, the post processing commands that are common to TEXLESP-S and TEXGAP3D are used. In this example, we calculate displacements, stresses and strains for load step 3. The data set shown in Figure 10 also produces a deflected shape plot.

# References

1. Oden, J. T., Finite Elements of Nonlinear Continua. New York: McGraw-Hill (1972).

2. Cescotto, S. and Fonder, G., "A Finite Element Approach for Large Strains of Nearly Incompressible Rubber-Like Materials," Int. J. Solids and Structs., V. 15, 589 (1979).

3. Aly, A. S., "A Finite Element Analysis for Problems of Large Strain and Large Displacement," TICOM 81-14, Austin (1981).

4. Miller, T. H., "A Finite Element Study of Instabilities in Rubber Elasticity," TICOM 83-2, Austin (1983).

5. Hibbitt, H. D., Karlessen, B. and Sorensen, P., "ABAQUS Users' Manual," H. K. S., Inc., Providence, RI (1984).

6. Becker, E., Collingwood, G. and Sato, T., "Users Manual for the TEXLESP Computer Code," AFRPL TR-84-085, V. 3 (1984).

7. Valanis, K. C. and Landel, R. F., J. Appl. Phys., V. 38, 2997-3002 (1967).

8. Ogden, R. W., "Large Deformation Isotropic Elasticity," Proc. Roy. Soc. Lond. A328 567-583 (1972).

9. Peng, S. T. J., "Nonlinear Multiaxial Finite Deformation Investigation of Solid Propellants," AFRPL TR-85-036 (1985).

10. Becker, E., Dunham, R. and Collingwood, G., "TEXGAP3D Users Manual," AFRPL TR-78-86 (1978).

11. Zienkiewicz, O. C., The Finite Element Method in Engineering Science. New York: McGraw-Hill (1974).

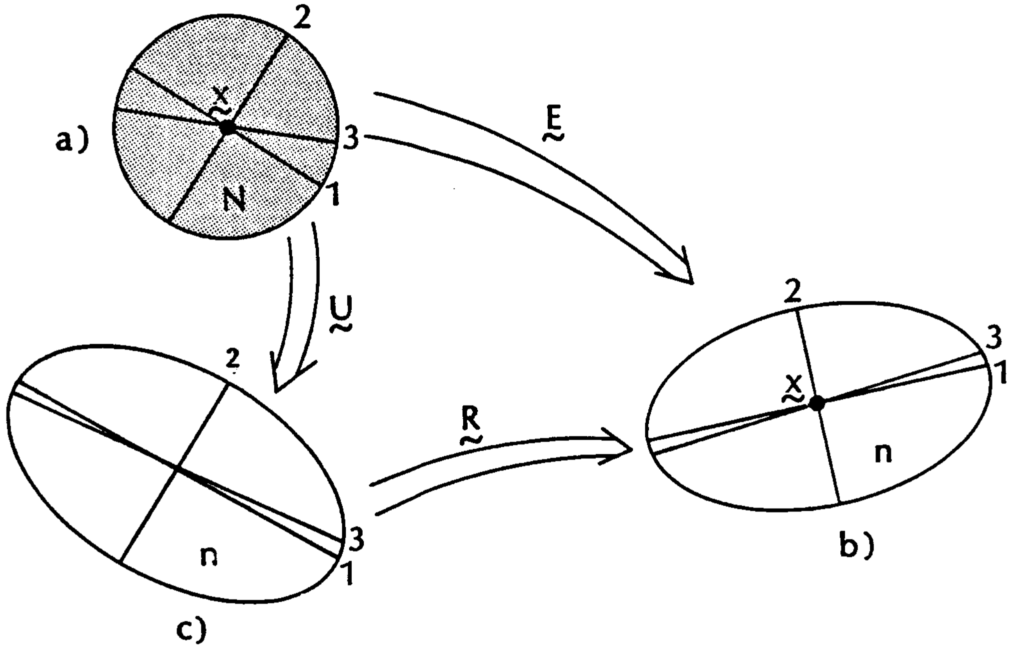


Figure 1. Deformation of a Neighborhood N into n

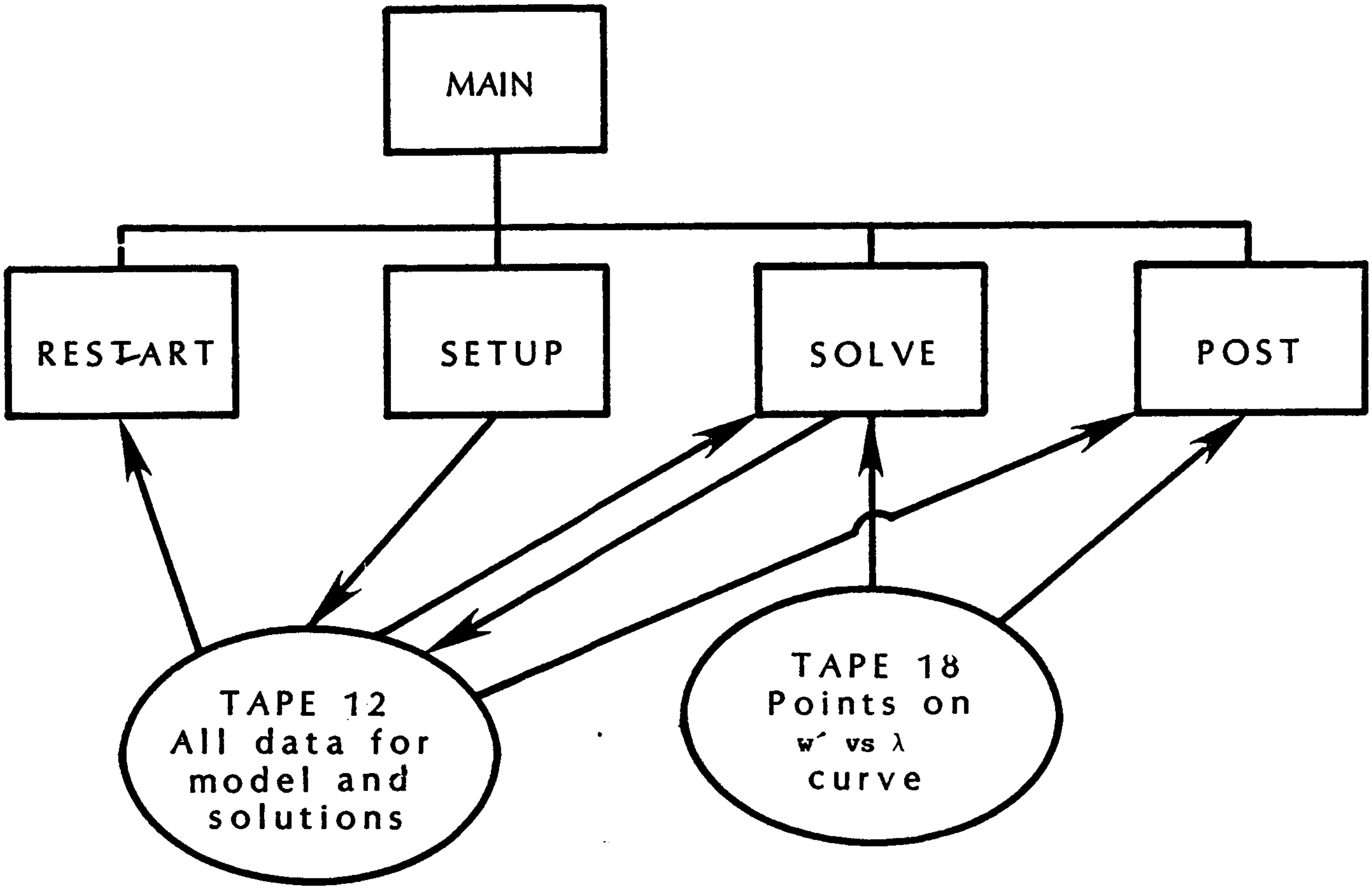


Figure 2. Schematic of TEXLESP-S

```
$TITLE
 SETINT
 SETUP
     (Material property data cards)
 END
     (Nodal point generation data cards)
 END
     (Element definition and boundary condition data
     cards)
 END
 SOLVE
 POST
     (Post processing commands)
 END
 STOP
```

Figure 3. Data Deck Structure for Complete Run

```
$TITLE
 RESTART
 SOLVE
 POST
     (Post processing commands)
 END
 STOP
```

Figure 4. Data Deck Structure for Post Processing Run

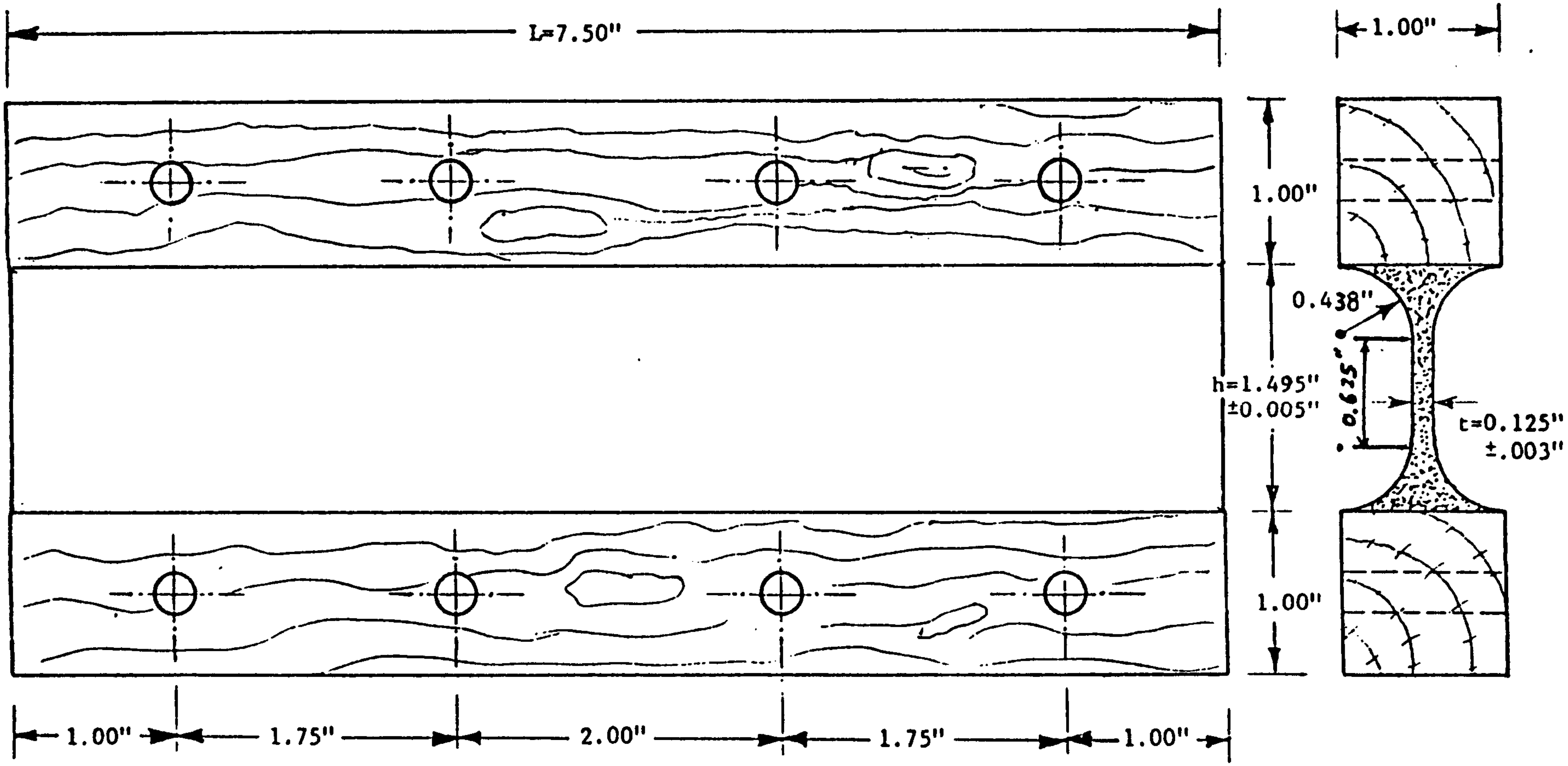


Figure 5. Thin Biaxial Test Specimen

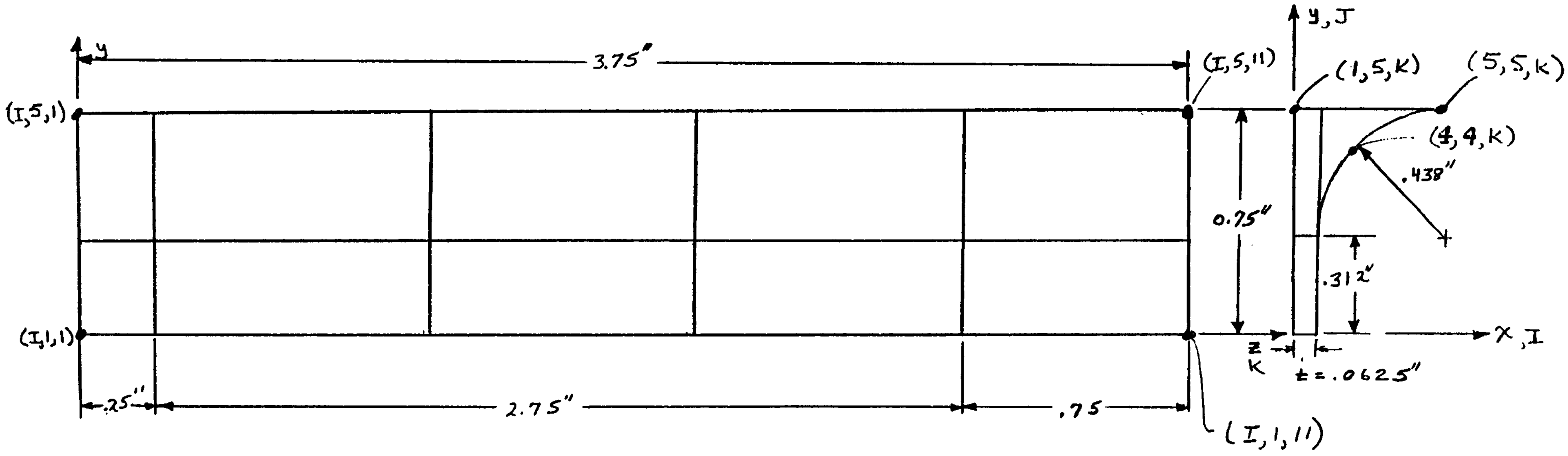
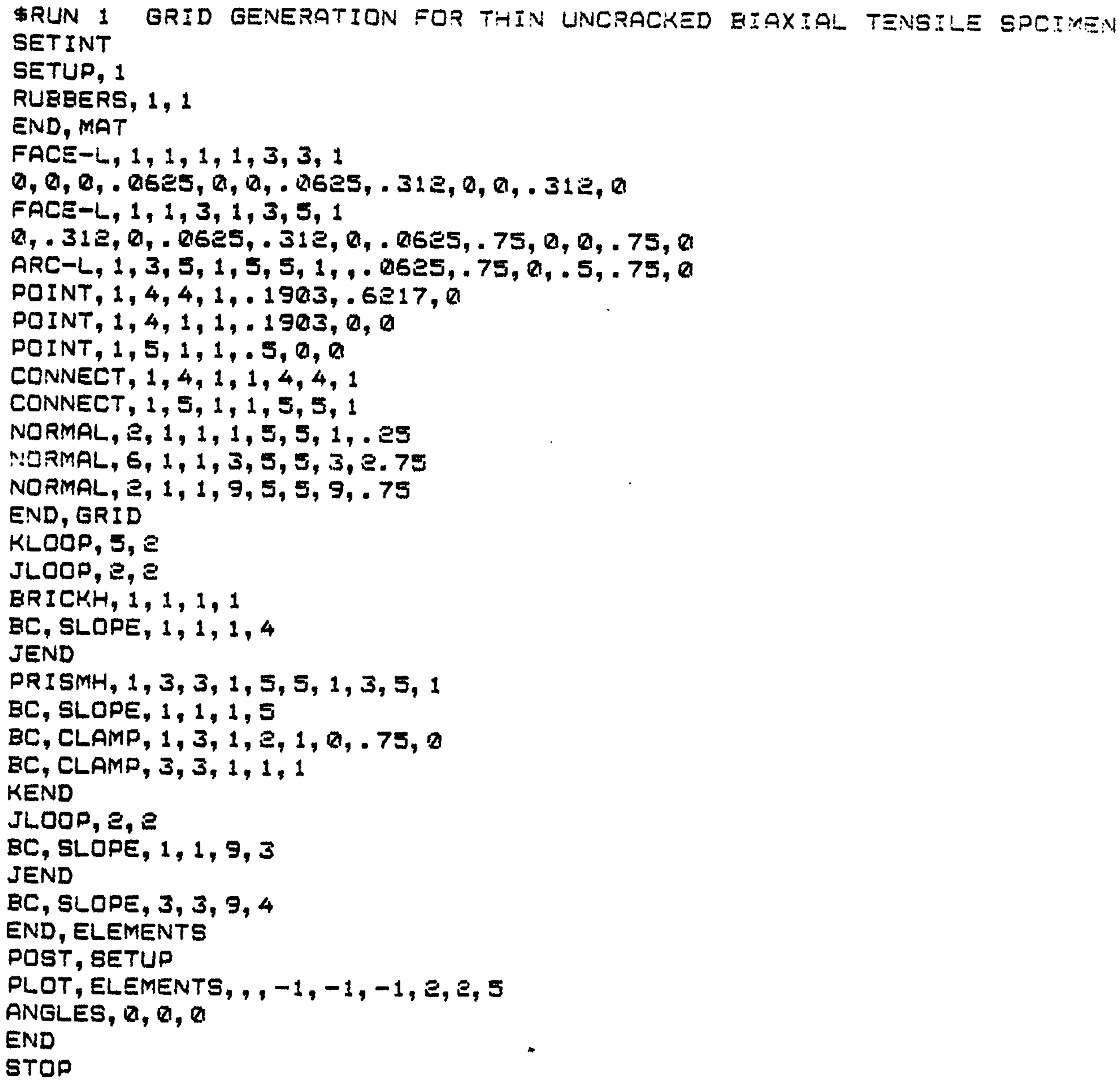


Figure 6. FEM of Thin Biaxial Test Specimen



```
$RUN 1  GRID GENERATION FOR THIN UNCRACKED BIAXIAL TENSILE SPCIMEN
SETINT
SETUP,1
RUBBERS,1,1
END,MAT
FACE-L,1,1,1,1,3,3,1
0,0,0,.0625,0,0,.0625,.312,0,0,.312,0
FACE-L,1,1,3,1,3,5,1
0,.312,0,.0625,.312,0,.0625,.75,0,0,.75,0
ARC-L,1,3,5,1,5,5,1,,.0625,.75,0,.5,.75,0
POINT,1,4,4,1,.1903,.6217,0
POINT,1,4,1,1,.1903,0,0
POINT,1,5,1,1,.5,0,0
CONNECT,1,4,1,1,4,4,1
CONNECT,1,5,1,1,5,5,1
NORMAL,2,1,1,1,5,5,1,.25
NORMAL,6,1,1,3,5,5,3,2.75
NORMAL,2,1,1,9,5,5,9,.75
END,GRID
KLOOP,5,2
JLOOP,2,2
BRICKH,1,1,1,1
BC,SLOPE,1,1,1,4
JEND
PRISMH,1,3,3,1,5,5,1,3,5,1
BC,SLOPE,1,1,1,5
BC,CLAMP,1,3,1,2,1,0,.75,0
BC,CLAMP,3,3,1,1,1
KEND
JLOOP,2,2
BC,SLOPE,1,1,9,3
JEND
BC,SLOPE,3,3,9,4
END,ELEMENTS
POST,SETUP
PLOT,ELEMENTS,,,-1,-1,-1,2,2,5
ANGLES,0,0,0
END
STOP
```

Figure 7. Data Deck for Grid Generation Run

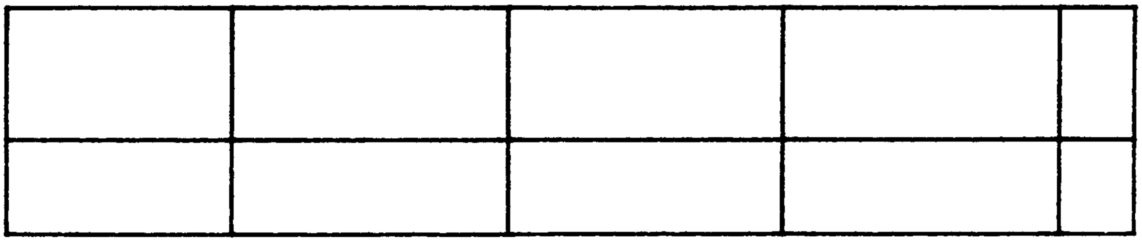


Figure 8. Grid Plot of FEM

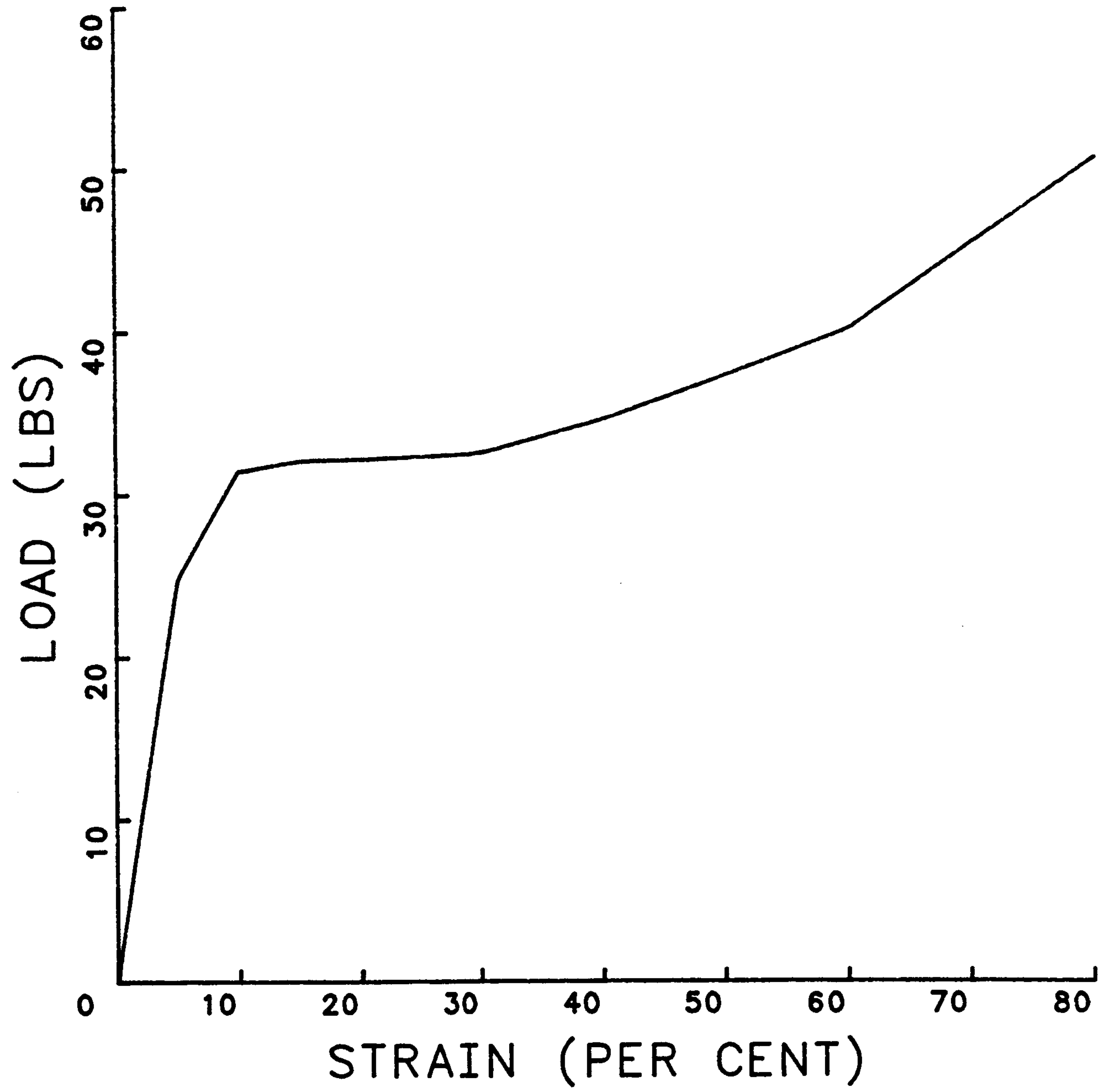
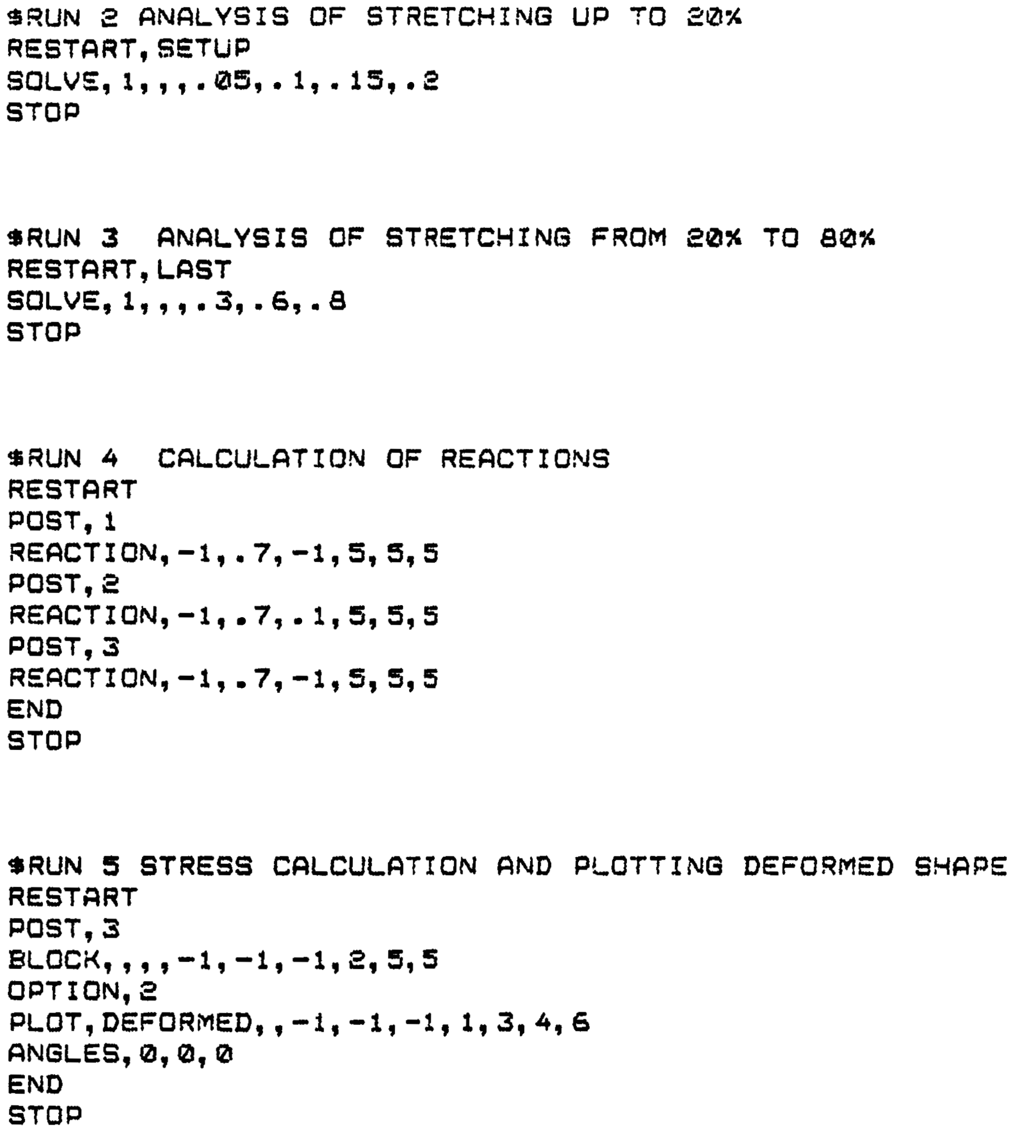


Figure 9. Load Deflection Curve



```
$RUN 2 ANALYSIS OF STRETCHING UP TO 20%
RESTART,SETUP
SOLVE,1,,,.05,.1,.15,.2
STOP


$RUN 3  ANALYSIS OF STRETCHING FROM 20% TO 80%
RESTART,LAST
SOLVE,1,,,.3,.6,.8
STOP


$RUN 4  CALCULATION OF REACTIONS
RESTART
POST,1
REACTION,-1,.7,-1,5,5,5
POST,2
REACTION,-1,.7,.1,5,5,5
POST,3
REACTION,-1,.7,-1,5,5,5
END
STOP


$RUN 5 STRESS CALCULATION AND PLOTTING DEFORMED SHAPE
RESTART
POST,3
BLOCK,,,,-1,-1,-1,2,5,5
OPTION,2
PLOT,DEFORMED,,-1,-1,-1,1,3,4,6
ANGLES,0,0,0
END
STOP
```

Figure 10. Data Decks for Restart Runs

# APPENDIX

## CALCULATION OF ELEMENT MATRICES

For a Rivlin polynomial material, the contribution of an element to the internal energy of the body is

$$I = \int_{\Omega^e} u(I_1, I_2, J) + P(J-1)\ dv$$

where the energy has been augmented to include the incompressibility constraint $(J-1 = 0)$. The contribution of the element to the right-hand side (or load vector) is $-\delta I$ . The variation in the energy is obtained by varying all the nodal displacement variables in the element $(x_i^\alpha)$ and the pressure variables $(p^\gamma)$. Therefore,

$$-\delta I = - \int_{\Omega^e} \{[u,_k \frac{\partial I_k}{\partial x_i^\alpha} + p \frac{\partial J}{\partial x_i^\alpha}]\delta x_i^\alpha + M^\gamma (J-1)\delta p^\gamma\} dv$$

where

$$u,_k \equiv \frac{\partial u}{\partial I_k} \qquad \left(\text{note: } I_3 \text{ denotes } J \equiv \det(\underset{\sim}{F})\right)$$

$M^\gamma$ = finite element interpolators for the pressure.

The contribution of the element to the tangent stiffness matrix is $d(\delta I)$ and is given by

$$d(\delta I) = \frac{\partial \delta I}{\partial x_j^\beta} dx_i^\beta + \frac{\partial \delta I}{\partial p^n} dp^n$$

$$= \int_{\Omega^e} \{[u,_{k\ell} \frac{\partial I_k}{\partial x_i^\alpha} \frac{\partial I_\ell}{\partial x_j^\beta} + u,_k \frac{\partial^2 I_k}{\partial x_i^\alpha \partial x_j^\beta} + p \frac{\partial^2 J}{\partial x_i^\alpha \partial x_j^\beta}] \delta x_i^\alpha dx_j^\beta$$

$$+ M^n \frac{\partial J}{\partial x_i^\beta} \delta x_i^\beta dp^n + M^\gamma \frac{\partial J}{\partial x_j^\beta} \delta p^\gamma dx_j^\beta\} dv \quad .$$

The first and second derivatives of the energy density function ($u,_k$ and $u,_{k\ell}$) can be obtained explicitly from the polynomial form of the energy function for Rivilin materials or as outlined in section 2.2 for materials defined by a $\omega'$ vs. $\lambda$ curve.

The invariants are complicated functions of the nodal point displacements ($x_i^\beta$) and derivatives of the invariants are best obtained by considering the derivatives of the deformation gradient ($\underset{\sim}{F}$), the left deformation tensor ($\underset{\sim}{B}$) and using the chain rule. Substituting the finite element interpolation of the nodal displacements into the definition of the deformation gradient gives

$$F_{mn} = \frac{\partial x_m}{\partial X_n}$$

$$= \frac{\partial N^\gamma}{\partial X_n} x_m^\gamma$$

$$= N_n^\gamma \, x_m^\gamma \qquad \left(\text{where} \quad N_n^\gamma \equiv \frac{\partial N^\gamma}{\partial X_n}\right) \quad .$$

The first and second derivatives of $\underset{\sim}{F}$ are:

$$\frac{\partial F_{mn}}{\partial x_i^\alpha} = N_n^\alpha \delta_{im}$$

$$\frac{\partial^2 F_{mn}}{\partial x_i^\alpha \partial x_j^\beta} = 0$$

where $\delta_{im}$ is the Kronecker delta function.

The left deformation tensor is defined by

$$B_{mn} = F_{mp} F_{np} \quad .$$

The first and second derivatives of $\underset{\sim}{B}$ are

$$\frac{\partial B_{mn}}{\partial x_i^\alpha} = F_{mp} \frac{\partial F_{np}}{\partial x_i^\alpha} + F_{np} \frac{\partial F_{mp}}{\partial x_i^\alpha}$$

$$= (F_{mp} \delta_{in} + F_{np} \delta_{im}) N_p^\alpha$$

$$\frac{\partial^2 B_{mn}}{\partial x_i^\alpha \partial x_j^\beta} = (\delta_{jm}\delta_{in} + \delta_{jn}\delta_{im}) N_p^\alpha N_p^\beta$$

$$= (\delta_{jm}\delta_{in} + \delta_{jn}\delta_{im}) \delta_{pq} N_p^\alpha N_q^\beta \quad .$$

The derivatives of the invariants $(I_1, I_2, J)$ can now be determined:

$$I_1 \equiv \delta_{mn} B_{mn}$$

$$\frac{\partial I_1}{\partial x_i^\alpha} = \delta_{mn} \frac{\partial B_{mn}}{\partial x_i^\alpha}$$

$$= 2F_{ip} N_p^\alpha$$

$$= N^{\alpha}_{p}\bar{B}^{1}_{pi}$$

$$\frac{\partial^2 I_1}{\partial x^{\alpha}_{i}\partial x^{\beta}_{j}} = \delta_{mn}\frac{\partial^2 B_{mn}}{\partial x^{\alpha}_{i}\partial x^{\beta}_{j}}$$

$$= 2\delta_{ij}\delta_{pq}\; N^{\alpha}_{p}N^{\beta}_{q}$$

$$= H^{1}_{ijpq}N^{\alpha}_{p}N^{\beta}_{q} \quad .$$

The second invariant is more complicated.

Let $(\underset{\sim}{B}^2)_{mn} = B_{mr}B_{rn}$

then

$$\bar{I}_1 \equiv \mathrm{tr}(\underset{\sim}{B}^2)$$

$$\bar{I}_1 = \delta_{mn}B_{mr}B_{rn}$$

$$= B_{mr}B_{rm} \quad .$$

The derivatives of $\bar{I}_1$ are :

$$\frac{\partial \bar{I}_1}{\partial x^{\alpha}_{i}} = 2B_{mr}\frac{\partial B_{mr}}{\partial x^{\alpha}_{i}}$$

$$= 2B_{mr}(F_{mp}\delta_{ir} + F_{rp}\delta_{im})N^{\alpha}_{p}$$

$$= 4B_{im}F_{mp}N^{\alpha}_{p}$$

$$\frac{\partial \bar{I}_1}{\partial x^{\alpha}_{i}\partial x^{\beta}_{j}} = 4[F_{mp}\frac{\partial B_{im}}{\partial x^{\beta}_{j}} + B_{im}\frac{\partial F_{mp}}{\partial x^{\beta}_{j}}]N^{\alpha}_{p}$$

$$= 4[F_{mp}(F_{iq}\delta_{jm} + F_{mq}\delta_{ji}) + B_{im}\delta_{jm}\delta_{pq}]N^{\alpha}_{p}N^{\beta}_{q}$$

$$= 4[F_{jp}F_{iq} + \delta_{ij}C_{pq} + B_{ij}\delta_{pq}]N^{\alpha}_{p}N^{\beta}_{q}$$

where $\underset{\sim}{C} \equiv \underset{\sim}{F}^T\underset{\sim}{F}$ is the right deformation tensor.

The second invariant is defined by

$$I_2 = \frac{1}{2}(I_1^2 - \bar{I}_1)$$

and its derivatives are:

$$\begin{aligned}
\frac{\partial I_2}{\partial x_i^\alpha} &= \frac{1}{2}\left(2I_1 \frac{\partial I_i}{\partial x_i^\alpha} - \frac{\partial \bar{I}_i}{\partial x_i^\alpha}\right) \\
&= I_1(2F_{ip}N_p^\alpha) - 2B_{im}F_{mp}N_p^\alpha \\
&= 2(F_{ip}I_1 - B_{im}F_{mp})N_p^\alpha \\
&= 2F_{mp}(\delta_{im}I_1 - B_{im})N_p^\alpha \\
&= N_p^\alpha \bar{B}_{pi}^2
\end{aligned}$$

$$\begin{aligned}
\frac{\partial^2 I_2}{\partial x_i^\alpha \partial x_j^\beta} &= \frac{\partial I_1}{\partial x_j^\beta}\frac{\partial I_1}{\partial x_i^\alpha} + I_1 \frac{\partial^2 I_1}{\partial x_i^\alpha \partial x_j^\beta} - \frac{1}{2}\frac{\partial^2 \bar{I}_i}{\partial x_i^\alpha \partial x_j^\beta} \\
&= [4F_{jq}F_{ip} + 2I_1\delta_{ij}\delta_{pq} - 2(F_{jp}F_{iq} \\
&\qquad + \delta_{ij}C_{pq} + B_{ij}\delta_{pq})]N_p^\alpha N_q^\beta \\
&= 2[2F_{ip}F_{jq} - F_{jp}F_{iq} + (I_1\delta_{ij} - B_{ij})\delta_{pq} \\
&\qquad - C_{pq}\delta_{ij}]N_p^\alpha N_q^\beta \\
&= H_{ijpq}^2 N_p^\alpha N_q^\beta
\end{aligned}$$

The determinate of $\underset{\sim}{F}$ is defined as

$$J \equiv \det(F) = F_{11}\hat{C}_{11} + F_{12}\hat{C}_{12} + F_{13}\hat{C}_{13}$$

where $\hat{C}_{ij}$ is the cofactor of $F_{ij}$ .

The first derivative of $J$ is

$$\frac{\partial J}{\partial x_i^\alpha} = \hat{C}_{ip} N_p^\alpha$$

$$= N_p^\alpha J (\underset{\sim}{F}^{-1})_{pi}$$

$$= N_p^\alpha \bar{B}_{pi}^3$$

where $\underset{\sim}{F}^{-1}$ is the inverse of the deformation gradient. In order to take the second derivatives of $J$ , it is necessary to write the first derivatives explicitly.

$$\frac{\partial J}{\partial x_1^\alpha} = N_1^\alpha (F_{22}F_{33} - F_{23}F_{32}) + N_2^\alpha (F_{23}F_{31} - F_{21}F_{33}) + N_3^\alpha (F_{21}F_{32} - F_{22}F_{31})$$

$$\frac{\partial J}{\partial x_2^\alpha} = N_1^\alpha (F_{13}F_{32} - F_{12}F_{33}) + N_2^\alpha (F_{11}F_{33} - F_{13}F_{31}) + N_3^\alpha (F_{12}F_{31} - F_{11}F_{32})$$

$$\frac{\partial J}{\partial x_3^\alpha} = N_1^\alpha (F_{12}F_{23} - F_{13}F_{22}) + N_2^\alpha (F_{13}F_{21} - F_{11}F_{23}) + N_3^\alpha (F_{11}F_{22} - F_{12}F_{21})$$

The second derivatives are:

$$\frac{\partial^2 J}{\partial x_i^\alpha \partial x_j^\beta} = 0 \quad \text{if} \quad i = j$$

otherwise,

$$\frac{\partial^2 J}{\partial x_1^\alpha \partial x_2^\beta} = N_1^\alpha(F_{33}N_2^\beta - F_{32}N_3^\beta) + N_2^\alpha(F_{31}N_3^\beta - F_{33}N_1^\beta) + N_3^\alpha(F_{32}N_1^\beta - F_{31}N_2^\beta)$$

$$\frac{\partial^2 J}{\partial x_1^\alpha \partial x_3^\beta} = N_1^\alpha(F_{22}N_3^\beta - F_{23}N_2^\beta) + N_2^\alpha(F_{23}N_1^\beta - F_{21}N_3^\beta) + N_3^\alpha(F_{21}N_2^\beta - F_{22}N_1^\beta)$$

$$\frac{\partial^2 J}{\partial x_2^\alpha \partial x_3^\beta} = N_1^\alpha(F_{13}N_2^\beta - F_{12}N_3^\beta) + N_2^\alpha(F_{11}N_3^\beta - F_{13}N_1^\beta) + N_3^\alpha(F_{12}N_1^\beta - F_{11}N_2^\beta)$$

$$\frac{\partial^2 J}{\partial x_i^\alpha \partial x_j^\beta} = - \frac{\partial^2 J}{\partial x_j^\alpha \partial x_i^\beta} \quad ; \quad i \neq j$$

or

$$\frac{\partial^2 J}{\partial x_i^\alpha \partial x_j^\beta} = H^3_{ijpq} \; N_p^\alpha N_q^\beta$$

Let $R_i^\alpha$ denote the component of the elemental right-hand side vector in the $i^{th}$ direction for node $\alpha$ and $R^\gamma$ denote the component of the vector corresponding to the $\gamma^{th}$ pressure variable. Then, using the expressions for the derivatives of the invariants

$$R_i^\alpha = - \int_{\Omega^e} [u,_1\bar{B}^1_{pi} + u,_2\bar{B}^2_{pi} + (u,_J+p)\bar{B}^3_{pi}]N_p^\alpha \, dv$$

$$R^\gamma = - \int_{\Omega^e} M^\gamma(J - 1) \, dv$$

Similarly, let $K_{ij}^{\alpha\beta}$ denote the component of the tangent stiffness matrix coupling the degree of freedom in direction $i$ at node $\alpha$ with the degree-of-freedom in direction $j$ at node $\beta$, $K_i^{\alpha n}$ denote the component coupling the $n^{th}$ pressure degree of freedom with the degree-of-freedom in direction $i$ at node $\alpha$, and $K^{\gamma n}$ denote the component coupling the $\gamma^{th}$ and $n^{th}$ pressure degrees of freedom. The stiffness matrix is then

$$K_{ij}^{\alpha\beta} = \int_{\Omega^e} [u_{,k\ell}\, \bar{B}_{pi}^{k}\, \bar{B}_{qj}^{\ell} + u_{,k}\, H_{ijpq}^{k} + pH_{ijpq}^{3}] N_p^{\alpha} N_q^{\beta}\, dv$$

$$K_i^{\alpha n} = \int_{\Omega^e} M^n\, \bar{B}_{pi}^{3}\, N_p^{\alpha}\, dv$$

$$K^{\gamma n} = 0 \quad .$$